IMPRINT - A control program for the IMP printer

Features:

- Full bidirectional printing for maximum throughput.
- Unidirectional printing for maximum column alignment
- Double width characters.
- Full graphics printing mode.
- Self test to confirm printer operational.
- Recognises Form Feed character.
- Responds to horizontal Tab character.

This document should be read in conjunction with the manual supplied with the IMP printer.

Design philosophy: The aim has been to include those features (such as double width characters) that cannot be implemented in an external printer driver.

Some IMPrint / Nas-Print comparisons.

Character set: An inverted question mark is now printed as the rubout character.

Direction of print: The printer head is specified for continuous bi-directional printing. Previously, problems had sometimes been encountered with continuous bi-directional printing of rub-out characters, so Nas-Print returned the head if a line contained more than 40 printing characters. As the dense rub-out character has been changed IMPrint offers full bi-directional printing or undirectional printing.

Busy line: Unchanged.

Strap options: Unchanged.

Error light: This performs the same functions as before - it lights if a transmission error occurs or the internal buffer overflows - and also has an additional function, see the section on printing a graphics line.
The Error light (if lit) is automatically extinguished whenever the internal buffer is empty.

Control characters recognised.

Hex code	Ascii	Function
02	Control/B	Set bi-directional print
03	Control/C	Set unidirectional print
04	Control/D	Start double width characters
05	Control/E	Start single width characters
08	Control/H	Backspace
09	Control/I	Horizontal Tab
0A	Control/J	Line feed
0C	Control/L	Form feed
0D	Control/M	Carriage return
1F	Control/_	Print graphics line

All other control codes are ignored.

Default options:

On power up the following options are set-
Bi-directional printing
Single width characters

Following a CR,LF or FF the following option is set
Single width characters

Self test mode.

On completion of the reset or power up procedure (when the "Error" LED is extinguished) IMPrint checks the line feed switch. If it finds it on it automatically prints the available character set four times, both in single width and in double width characters. This can be used at any time by pressing the line feed switch and briefly depressing the Reset switch. If the print head is at the right hand side wait until it has been moved to the left and the Error LED extinguished before releasing the line feed switch.

This provides a copy of the available character set and confirms that the major part of the IMP is operational.

Printing.

Incoming characters are stored in an internal buffer in the IMP. A line of characters is only printed when a CR LF or FF character is encountered. While the IMP is printing a line it will still accept characters provided that there is space available in its internal buffer. As before imminent overflow is signaled by the Busy line, and actual overflow causes the "Error" light to be lit. The action of the various control codes is explained in more detail below.

Code 02 (Control/B) Set bi-directional print

This code causes the IMP to print successive lines in alternate directions so maximising the the throughput rate of the printer. It takes effect immediately on receipt and is not queued in the internal buffer with the other characters.

Code 03 (Control/C) Set unidirectional print

This code causes the IMP to print successive lines in the same direction. The throughput rate of the IMP is lowered as the head is returned after each line is printed, but it offers the best possible column alignment. As with code 04 it takes immediate effect.

Code 04 (Control/D) Double width characters

This code causes the IMP to print all subsequent characters in a double width form. This option is automatically reset at the end of a line.

Code 05 (Control/E) Single width characters

The code causes the IMP to print all subsequent characters in single width form. This is the default condition at the start of any line.

Code 08 (Control/H) Backspace

This code causes the last character received to be deleted.
NOTE: Backspace only works within a line, and it will not backspace over a CR, LF or FF character.

Code 09 (Control/I) Horizontal tab

This causes the printer to insert spaces into the line until the next tab-stop is reached. Tab-stops exist implicitly at every eighth column. ie 8,16,24, etc . This is identical to the tab function within the CP/M disc operating system.

Code 0A (Control/J) Line feed

This code causes any preceeding characters in the buffer to be printed and the paper is advanced by one line.

Code 0C (Control/L) Form feed

This code is similar in action to the line feed, but results in the paper being advanced by six lines.

Code 0D (Control/M) Carriage return

This code causes any preceeding characters in the buffer to be printed. If the optional internal strap is set the paper is also advanced by one line.

Code 1F (Control/_) Graphics line

Any characters preceeding this code will be printed and the head will be positioned to the left hand side of the print area at the start of a fresh line if it is not already there. IMPrint now assumes that the next 760 bytes it receives will be the dot patterns it will use to drive the head during the next traverse of the printing area. ie The first byte following the 1F code will define the first vertical column of dots, (bit 6 = top dot bit 0 = bottom dot), the next byte the next column, and so on. IMPrint waits until it has 760 bytes in the buffer and then proceeds to print them across the page. Note that this is done solely on character count and no special terminating character or character sequence is required, or recognised. On completion of the traverse the head is returned to the left hand side and a partial line feed is performed. IMPrint returns to its ´single-width character´ mode. If a subsequent graphics line is required it is necessary to repeat the 1F character followed by another 760 bytes of dot pattern. In this mode the buffer works in the same manner as before, accepting bytes at any time provided space is available (signalled by the busy line).

NOTE: To meet the head specification, if a dot has been printed in one column it must be off in the following column. ie To print a horizontal line using bit 0 the following dot pattern is acceptable ...01 00 01 00 01 00 01 00... . The sequence ...01 01 01 01 01... is not. IMPrint ensures that this rule is not infringed by resetting any illegal bits in the dot patterns. If it has to reset any dots (which is done as it prints the line) it lights the "Error" light. This is extinguished once the line has been printed.

GRAPHICS MODE FROM BASIC

If a very long line is PRINTed, (as would be the case with a graphics line), some Basics will automatically insert a "carriage-return line feed" character pair every n characters, where n has been set by the WIDTH n command. In Microsoft Basic this should be switched off by giving a WIDTH 255 command.

The "X" command on a Nascom running Nas-Sys 1 does not output nulls. (Corrected in Nas-Sys 3). If you use "X" to turn on your printer output then "U" should be used instead, and the UOUT reflection set to use SRLX. (ie set 0C77 to DF 6F C9).

```
                                        .Z80
                                        TITLE   IMPRINT V1.0  08-01-80
                                        SUBTTL  Address/Constant Equates
                                ; Move to buffer for programming
 0000'   21 0090'                       LD      HL,PROG
 0003'   11 2000                        LD      DE,2000H
 0006'   01 0800                        LD      BC,2048
 0009'   ED B0                          LDIR
 000B'   C9                             RET
                                        PAGE    60
```

Address/Constant Equates

```
 000C´

                                ;  Control program for Nascom Imp printer
                                ;
                                ;       Version 1.0   08-01-81
                                ;       (C)  Copyright   D.W.Parkinson  1980
                                ;

                                ;               Port Addresses

 0018                           UARTDT  EQU     18H             ;UART Data
 0028                           UARTST  EQU     28H             ;UART Status
 0033                           PIOBC   EQU     33H             ;PIO B Control port
 0031                           SOLEN   EQU     31H             ;Solenoid control port
 0032                           PIOAC   EQU     32H             ;PIO A-side Control port
 0030                           SWITCH  EQU     30H             ;Switch etc port
 001F                           MASK    EQU     1FH             ;I/O mask for port A
 0006                           MASKLS  EQU     6               ;Mask for limit switches

                                ; PIO  Printer control Bit numbers

 0000                           AUTOLF  EQU     0               ;Auto LF Strap
 0001                           RHLMT   EQU     1               ;Right Hand limit switch
 0002                           LHLMT   EQU     2               ;Left Hand limit switch
 0003                           LFSW    EQU     3               ;Line feed push button
 0004                           ONLINE  EQU     4               ;Online switch
 0005                           LFSTEP  EQU     5               ;Line feed stepper motor
 0006                           BUSY    EQU     6               ;Busy line
 0007                           RSLED   EQU     7               ;Reset LED

                                ;             Constants

 FFE4                           SETBI   EQU     2-´ ´+2         ;Modified Set bidirect. print
 FFE5                           SETUNI  EQU     3-´ ´+2         ;  "  Set unidirectional print
 FFEA                           MBS     EQU     8-´ ´+2         ;  "  Backspace
 FFEB                           MTAB    EQU     9-´ ´+2         ;  "  Tab character
 FFEC                           MLF     EQU     0AH-´ ´+2       ;  "  LF
 FFEE                           MFF     EQU     0CH-´ ´+2       ;  "  Form Feed
 FFEF                           MCR     EQU     0DH-´ ´+2       ;  "  CR
 FFE7                           MDWOFF  EQU     5H-´ ´+2        ;  "  Double width off
 FFE6                           MDWON   EQU     4H-´ ´+2        ;  "  Double Width on
 0001                           ONGRAF  EQU     1FH-´ ´+2       ;  "  Enter graphics mode
 0062                           DWON    EQU     62H             ;Double width on
 0063                           DWOFF   EQU     63H             ;Double width off
 0064                           TAB     EQU     64H
 0065                           LF      EQU     65H             ;Line feed code
 0066                           CR      EQU     66H             ;Carriage return code
 0067                           FF      EQU     67H             ;Form Feed code
 0001                           EMPTY   EQU     1               ;Empty character
 001C                           ERRMSK  EQU     1CH             ;Status error mask

                                ;           Bit numbers of Flag byte

 0000                           GRAPH   EQU     0               ;Graphics mode flag
 0001                           DWIDTH  EQU     1               ;Double width (printing)
 0002                   POLL    EQU     2               ;Poll flag
 0003                   BIDIR   EQU     3               ;Uni/Bi-directional
 0004                   REVPRT  EQU     4               ;Set if head starts at RHS
 0005                   PRTING  EQU     5               ;Set for printing a line
                        ;       EQU     6               ;Dummy use BUSY above
 0007                   SPACE   EQU     7

                        ;          Variables

                                .PHASE  2800H           ;Start of RAM
 2800                   RAM:    DEFS    1               ;For 0C3H
 2801                   NMIREF: DEFS    2               ;NMI reflection
 2803                   COUNT:  DEFS    1               ;NMI counter
 2804                   FLAG:   DEFS    1               ;Flag byte
 2805                   HEDDAT: DEFS    1               ;Current solenoid data
 2806                   LASTDT: DEFS    1               ;Last col of dots
 2807                   GCOUNT: DEFS    2               ;Graphics counter
 2809                   INPTR:  DEFS    2               ;Input pointer
 280B                   PLINE:  DEFS    2               ;Start of current/next line
 280D                           DEFS    38              ;Stack area
 2833                   STACK:
 2833                           DEFS    1               ;For Null marker
 2834                   PRTBUF: DEFS    80              ;Print Buffer
 2884                   BUFST:
 2BFF                   BUFEND  EQU     RAM+1023
                                .LIST
                                PAGE
 0000
                                ; Restart routines

                                ;
                                ;    RST 0 - Reset initialisation routine
                                ;
 0000     31 2833               RESET:  LD      SP,STACK         ;Set stack pointer
 0003     C3 026A                       JP      INIT             ;Go to routine
 0006     FF FF                         DEFB    0FFH,0FFH        ;Filler
                                ;
                                ; RST INCHL - Increment up the buffer
                                ;
 0008     23                    INCHL:  INC     HL               ;Up the buffer
 0009     7E                            LD      A,(HL)           ;Check for 0 marker
 000A     B7                            OR      A
 000B     C0                            RET     NZ               ;Return not there
 000C     21 2885                       LD      HL,BUFST+1       ;Reset HL to start
 000F     C9                            RET
                                ;
                                ; RST DECHL - Move back down the buffer
                                ;
 0010     2B                    DECHL:  DEC     HL               ;Back
 0011     7E                            LD      A,(HL)           ;Check for limit marker
 0012     B7                            OR      A
 0013     C0                            RET     NZ               ;Return not there
 0014     21 2BFE                       LD      HL,BUFEND-1      ;"Wrap round"
 0017     C9                            RET

                                ;
                                ; RST LEDON - Turn on "Error" LED
                                ;
 0018     DB 30                 LEDON:  IN      A,(SWITCH)       ;Get current states
 001A     CB BF                         RES     RSLED,A          ;Turn on LED
 001C     D3 30                         OUT     (SWITCH),A       ;Light it
 001E     C9                            RET
 001F     FF                            DEFB    -1               ;Filler
                                ;
                                ; Reset NMI reflection to POLL
                                ;
 0020     E5                    RESNMI: PUSH    HL               ;Preserve HL
 0021     21 016F                       LD      HL,POLLU
 0024     22 2801                       LD      (NMIREF),HL
 0027     E1                            POP     HL
 0028     DD CB 00 B6                   RES     BUSY,(IX)        ;Clear the busy flag
 002C     C9                            RET
 002D     FF FF FF                      DEFB    0FFH,0FFH,0FFH   ;Filler
                                ;
                                ; Check limit switches, Return Z=Print area
                                ;                                NZ=one end
 0030     DB 30                 CHLMSW: IN      A,(SWITCH)       ;Read switches
 0032     2F                            CPL
 0033     E6 06                         AND     MASKLS           ;Mask off
 0035     C9                            RET                      ;Z=In print area
 0036     FF FF                         DEFB    0FFH,0FFH
                                ;
                                ; Toggle REVPRT bit   On if at RHS
```

```
                                    ; Can´t Read-Modify-Write byte because of
                                    ;   interaction with Interrupts
                                    ;
  0038    DD CB 00 66               CHSIDE: BIT     REVPRT,(IX)     ;On?
  003C    28 05                             JR      Z,SETRVP        ;No,set on
  003E    DD CB 00 A6                       RES     REVPRT,(IX)     ;Clear bit
  0042    C9                                RET
  0043    DD CB 00 E6               SETRVP: SET     REVPRT,(IX)
  0047    C9                                RET

                                    ;  Turn the Head Motor on

  0048    DD CB 00 F6               MOTON:  SET     BUSY,(IX)       ;Say Busy
  004C    F5                                PUSH    AF              ;Save AF...
  004D    C5                                PUSH    BC              ;...and BC
  004E    06 08                             LD      B,8             ;Loop count
  0050    AF                                XOR     A               ;Initialise A
  0051    32 2805                           LD      (HEDDAT),A      ;Set it
  0054    0E 0A                     MOTON0: LD      C,10            ;Wait 5ms
  0056    76                        MOTON1: HALT
  0057    0D                                DEC     C
  0058    20 FC                             JR      NZ,MOTON1
  005A    EE 80                             XOR     80H             ;Toggle bit
  005C    32 2805                           LD      (HEDDAT),A      ;Update it
  005F    10 F3                             DJNZ    MOTON0          ;Loop if more
  0061    C1                                POP     BC              ;Reset registers
  0062    F1                                POP     AF
  0063    C9                                RET                     ;Done
                                            SUBTTL NMI Routines
                                            REPT    66H-$
                                            DEFB    0FFH            ;Filler
                                            ENDM
  0064    FF                  +             DEFB    0FFH            ;Filler
  0065    FF                  +             DEFB    0FFH            ;Filler
                                            PAGE
```

```
0066

                                ; This section of code is executed every time
                                ; there is an NMI.

0066    F5                      NMI:    PUSH    AF              ;Preserve flags
0067    3A 2805                         LD      A,(HEDDAT)      ;Get Head Data
006A    D3 31                           OUT     (SOLEN),A               ;Fire Solenoids
006C    E5                              PUSH    HL              ;Save HL
006D    21 2803                         LD      HL,COUNT        ;Point to COUNT
0070    35                              DEC     (HL)            ;Update it
0071    C3 2800                         JP      RAM             ;Then thru' NMI reflection

                                ; This section handles line feeds
                                ; Entered with Z set according to COUNT

0074    C2 016F                 LFEED:  JP      NZ,POLLU        ;Not time yet
0077    DB 30                           IN      A,(SWITCH)      ;Pulse the stepper motor
0079    CB EF                           SET     LFSTEP,A
007B    D3 30                           OUT     (SWITCH),A
007D    CB AF                           RES     LFSTEP,A
007F    D3 30                           OUT     (SWITCH),A
0081    36 0A                           LD      (HL),10         ;Reset NMI count
0083    D9                              EXX                     ;Get LF register set
0084    10 01                           DJNZ    LF0             ;More to the line?,yes=skip
0086    E7                              RST     RESNMI          ;Reset NMI reflection
0087    D9                      LF0:    EXX                     ;Reset the registers
0088    C3 016F                         JP      POLLU           ;Off to check the UART

                                ; This section waits for both limit switches
                                ; to be off and then starts the print routine

008B    F7                      WAIT:   RST     CHLMSW          ;Check limit switches
008C    C2 016F                         JP      NZ,POLLU        ;Still set - go Poll
008F    21 00D6                         LD      HL,PRINT        ;Change reflection
0092    22 2801                         LD      (NMIREF),HL
0095    D9                              EXX                     ;Set up the registers
0096    06 01                           LD      B,1
0098    21 2833                         LD      HL,PRTBUFF-1
009B    DD CB 00 66                     BIT     REVPRT,(IX)     ;Backwards?
009F    28 03                           JR      Z,WAIT0         ;No,skip
00A1    21 2884                         LD      HL,PRTBUFF+80
00A4    D9                      WAIT0:  EXX
00A5    21 2803                         LD      HL,COUNT        ;Reset HL
00A8    36 05                           LD      (HL),5          ;Reset count to 5
00AA    C3 016F                         JP      POLLU           ;Poll the UART

                                ; This section waits for the appropriate limit
                                ; switch and then stops the motor after 70ms

00AD    F7                      MSTOP:  RST     CHLMSW          ;Read switches
00AE    CA 016F                         JP      Z,POLLU         ;Still in print area
00B1    DD CB 00 66                     BIT     REVPRT,(IX)     ;Which test?
00B5    20 04                           JR      NZ,STARTR       ;Skip RHS wanted
00B7    CB 57                           BIT     LHLMT,A         ;At left?
```

```
 00B9     18 02                        JR      MSTOP0
 00BB     CB 4F             STARTR: BIT     RHLMT,A          ;At right?
 00BD     CA 016F           MSTOP0: JP      Z,POLLU          ;No,ignore
 00C0     E5                        PUSH    HL               ;There,so change reflection
 00C1     21 00CA                   LD      HL,MSTOP1
 00C4     22 2801                   LD      (NMIREF),HL
 00C7     E1                        POP     HL
 00C8     36 8C                     LD      (HL),140         ;Set time out count
                            ; NZ set at this point so can fall thru
 00CA     C2 016F           MSTOP1: JP      NZ,POLLU         ;Skip if not there yet
 00CD     3E 80                     LD      A,80H            ;Motor off
 00CF     32 2805                   LD      (HEDDAT),A
 00D2     E7                        RST     RESNMI           ;Normal NMI
 00D3     C3 016F                   JP      POLLU

                            ; This section prints a line

 00D6     C2 016F           PRINT:  JP      NZ,POLLU         ;On to Poll if still waiting
 00D9     36 01                     LD      (HL),1           ;Set count to drop thru again
 00DB     D9                        EXX                      ;Get "printing" registers
 00DC     05                        DEC     B                ;More cols?
 00DD     28 2E                     JR      Z,ENDCHR         ;Skip if end of char.
                            ; Get next column of dots
 00DF     CB 40                     BIT     0,B              ;Turn dots off?
 00E1     20 24                     JR      NZ,DOTOFF        ;Yes,skip
 00E3     1A                        LD      A,(DE)           ;Get next col. of dots
 00E4     DD CB 00 4E               BIT     DWIDTH,(IX)      ;Double width?
 00E8     28 12                     JR      Z,SWIDTH         ;No,skip
 00EA     CB 48                     BIT     1,B              ;All off?
 00EC     28 19                     JR      Z,DOTOFF         ;Yes,skip
 00EE     DD B6 02                  OR      (IX+LASTDT-FLAG);No,include previous
 00F1     4F                        LD      C,A              ;Temp save A
 00F2     1A                        LD      A,(DE)           ;Update LASTDT
 00F3     32 2806                   LD      (LASTDT),A
 00F6     78                        LD      A,B              ;Check for last Col.
 00F7     FE 06                     CP      6                ;(Set flags)
 00F9     79                        LD      A,C              ;(Reset A)
 00FA     28 0C                     JR      Z,DOTOFF+1       ;If so leave DE alone
 00FC     13                SWIDTH: INC     DE               ;On to next dots
 00FD     DD CB 00 66               BIT     REVPRT,(IX)      ;Backwards?
 0101     28 05                     JR      Z,DOTOFF+1       ;No,skip
 0103     1B                        DEC     DE
 0104     1B                        DEC     DE               ;Yes,backspace pointer
 0105     18 01                     JR      DTF
 0107     AF                DOTOFF: XOR     A                ;Clear A
 0108     32 2805           DTF:    LD      (HEDDAT),A       ;Set for next NMI
 010B     18 61                     JR      PRT99            ;Move on
                            ; End of character - do inter character gap
                            ; and set up for next character
 010D     D9                ENDCHR: EXX
 010E     36 05                     LD      (HL),5           ;Set inter-ch.-gap
 0110     DD CB 00 4E               BIT     DWIDTH,(IX)      ;Double width?
 0114     28 02                     JR      Z,ECO            ;No,skip
 0116     36 08                     LD      (HL),8           ;Yes,increase gap
 0118     D9                ECO:    EXX
 0119     2C                        INC     L                ;On to next character
```

```
011A    DD CB 00 66                  BIT     REVPRT,(IX)      ;Backwards?
011E    28 02                        JR      Z,EC1            ;No,skip
0120    2D                           DEC     L
0121    2D                           DEC     L
0122    B6                  EC1:     OR      (HL)             ;Load it setting flags
0123    20 0C                        JR      NZ,NOTEOL        ;Skip if more on line
                            ; End of print line
0125    DD CB 00 AE                  RES     PRTING,(IX)      ;Turn off print flag
0129    21 00AD                      LD      HL,MSTOP         ;Set auto time-out
012C    22 2801                      LD      (NMIREF),HL
012F    18 3D                        JR      PRT99
                            ; Another character
0131    17                  NOTEOL:  RLA                      ;Msb to Carry,0 to LSB
0132    DD CB 00 8E                  RES     DWIDTH,(IX)      ;Clear double width flag
0136    30 0D                        JR      NC,NTL0          ;Skip if single
0138    DD CB 00 CE                  SET     DWIDTH,(IX)      ;Else set it
013C    2C                           INC     L                ;(and skip repeat)
013D    DD CB 00 66                  BIT     REVPRT,(IX)
0141    28 02                        JR      Z,NTL0
0143    2D                           DEC     L
0144    2D                           DEC     L
0145    0F                  NTL0:    RRCA                     ;Reset A
0146    D6 02                        SUB     2                ;Remove Bias
0148    4F                           LD      C,A              ;Compute dot pattern..
0149    06 00                        LD      B,0              ;...address of next..
014B    07                           RLCA                     ;*2
014C    5F                           LD      E,A              ;...char.
014D    50                           LD      D,B
014E    EB                           EX      DE,HL
014F    29                           ADD     HL,HL            ;*4
0150    29                           ADD     HL,HL            ;*8
0151    ED 42                        SBC     HL,BC            ;*7
0153    DD 70 02                     LD      (IX+LASTDT-FLAG),B;Clear LASTDT
0156    01 0560                      LD      BC,TABLE         ;Base address
0159    DD CB 00 66                  BIT     REVPRT,(IX)      ;Backwards?
015D    28 03                        JR      Z,NTL1           ;No,skip
015F    01 0566                      LD      BC,TABLE+6       ;Yes,offset to top end
0162    09                  NTL1:    ADD     HL,BC            ;Index in
0163    06 0F                        LD      B,15             ;Set col count in B
0165    DD CB 00 4E                  BIT     DWIDTH,(IX)      ;If double...
0169    28 02                        JR      Z,NTL2
016B    06 1F                        LD      B,31             ;B=31
016D    EB                  NTL2:    EX      DE,HL            ;Reset registers
016E    D9                  PRT99:   EXX                      ;Reset the registers

                            ; This section Polls the UART if polling
                            ; is enabled. It will be disabled if the IMP
                            ; is off line or the polling routine has been
                            ; caught by an NMI.

016F    23                  POLLU:   INC     HL               ;Move onto FLAG
0170    CB 56                        BIT     POLL,(HL)        ;Poll the UART?
0172    CA 0243                      JP      Z,NMIEXT         ;No,exit the routine
0175    DB 28                        IN      A,(UARTST)       ;Get status byte
0177    17                           RLA                      ;Character ready?
0178    DA 0243                      JP      C,NMIEXT         ;No,return
```

```
                                ;
                                ; Character ready, accept it and check it and
                                ; and buffer bounds.
                                ;
017B    CB 96                           RES     POLL,(HL)       ;Don´t re-enter until finished
017D    2A 2809                         LD      HL,(INPTR)      ;Pick up input pointer
                                ;
0180    2F                              CPL                     ;Check for transmission error
0181    E6 1C                           AND     ERRMSK
0183    DB 18                           IN      A,(UARTDT)      ;(Read UART)
0185    28 03                           JR      Z,UARTO         ;Ok,skip
0187    DF                              RST     LEDON           ;Set error Led
0188    3E 7F                           LD      A,7FH           ;Set Error byte
018A    E6 7F                   UARTO:  AND     7FH             ;Strip parity bit
018C    DD CB 00 46                     BIT     GRAPH,(IX)      ;Graphics mode?
0190    20 4F                           JR      NZ,PUTGRF       ;Yes,save unmodified
0192    D6 20                           SUB     ´ ´             ;Printable?
0194    3C                              INC     A               ;(Adjust)
0195    3C                              INC     A
0196    D2 0239                         JP      NC,PUT          ;Yes,skip
                                ; Check possible control characters
0199    FE EF                           CP      MCR             ;Was it CR?
019B    CA 0223                         JP      Z,CRIN          ;Yes
019E    FE EC                           CP      MLF             ;LF?
01A0    CA 0237                         JP      Z,LFIN          ;Yes
01A3    FE EE                           CP      MFF             ;FF?
01A5    28 78                           JR      Z,FFIN
01A7    FE EA                           CP      MBS             ;Backspace?
01A9    28 59                           JR      Z,BSIN
01AB    FE EB                           CP      MTAB            ;Tab character?
01AD    28 6C                           JR      Z,TABIN
01AF    FE E6                           CP      MDWON           ;Double width on?
01B1    28 60                           JR      Z,SETDW
01B3    FE E7                           CP      MDWOFF
01B5    28 60                           JR      Z,CLRDW
01B7    FE E4                           CP      SETBI           ;Bidirectional on?
01B9    28 3D                           JR      Z,BION
01BB    FE E5                           CP      SETUNI          ;Unidirectional on?
01BD    28 3F                           JR      Z,UNION
01BF    FE 01                           CP      ONGRAF          ;Enter graphics mode?
01C1    20 79                           JR      NZ,UART99       ;No,ignore all others
                                ; Enter Graphics mode
01C3    DD CB 00 C6                     SET     GRAPH,(IX)      ;Set flag bit
01C7    E5                              PUSH    HL              ;Temp save HL
01C8    21 02F8                         LD      HL,760          ;Set Graphics count
01CB    22 2807                         LD      (GCOUNT),HL
01CE    E1                              POP     HL              ;Reset HL
01CF    7E                              LD      A,(HL)          ;Get last byte
01D0    FE 01                           CP      EMPTY           ;Nothing there?
01D2    28 68                           JR      Z,UART99        ;Ok,exit
01D4    FE 65                           CP      LF              ;Line feed?
01D6    28 64                           JR      Z,UART99        ;Ok,exit
01D8    FE 67                           CP      FF              ;Form feed?
01DA    28 60                           JR      Z,UART99        ;Ok,exit
01DC    07                              RLCA                    ;Graphics charcter?
01DD    38 5D                           JR      C,UART99        ;Ok,exit
```

```
01DF     18 56                   JR      LFIN           ;Else insert a LF.
                         ; Save a graphics character
01E1     F6 80           PUTGRF: OR      80H            ;Set msb
01E3     F5                      PUSH    AF             ;Save A
01E4     E5                      PUSH    HL             ;And HL
01E5     2A 2807                 LD      HL,(GCOUNT)    ;Check count
01E8     2B                      DEC     HL
01E9     22 2807                 LD      (GCOUNT),HL
01EC     7C                      LD      A,H
01ED     B5                      OR      L
01EE     E1                      POP     HL             ;(Reset HL)
01EF     20 04                   JR      NZ,GOING       ;Skip if not finished
01F1     DD CB 00 86             RES     GRAPH,(IX)     ;Else clear Graphics flag
01F5     F1              GOING:  POP     AF             ;Reset A
01F6     18 41                   JR      PUT            ;Save the byte
                         ; Bi/Uni directional
01F8     DD CB 00 9E     BION:   RES     BIDIR,(IX)
01FC     18 3E                   JR      UART99
01FE     DD CB 00 DE     UNION:  SET     BIDIR,(IX)
0202     18 38                   JR      UART99
                         ; Backspace
0204     7E              BSIN:   LD      A,(HL)         ;Check last byte
0205     FE 65                   CP      LF             ;Line feed or greater?
0207     30 33                   JR      NC,UART99      ;Yes,can´t backspace
0209     3D                      DEC     A              ;Empty?
020A     28 30                   JR      Z,UART99       ;Yes,no backspace
020C     36 01                   LD      (HL),EMPTY     ;Else flag empty
020E     D7                      RST     DECHL          ;Backspace pointer
020F     FD 23                   INC     IY             ;Update free count
0211     18 29                   JR      UART99         ;Exit
                         ; Double width
0213     3E 62           SETDW:  LD      A,DWON
0215     18 22                   JR      PUT
0217     3E 63           CLRDW:  LD      A,DWOFF
0219     18 1E                   JR      PUT
                         ; Tab character
021B     3E 64           TABIN:  LD      A,TAB
021D     18 1A                   JR      PUT
                         ; Process a Form Feed
021F     3E 67           FFIN:   LD      A,FF
0221     18 16                   JR      PUT
                         ; Process a carriage return
0223     7E              CRIN:   LD      A,(HL)         ;Check previous charac.
0224     FE 66                   CP      CR             ;CR?
0226     28 14                   JR      Z,UART99       ;Yes,ignore
0228     FE 65                   CP      LF             ;Line feed?
022A     28 10                   JR      Z,UART99       ;Yes,ignore
022C     3E 66                   LD      A,CR           ;Set CR code
022E     CD 0247                 CALL    STORE          ;Put in buffer
0231     DB 30                   IN      A,(SWITCH)     ;Check straps
0233     CB 47                   BIT     AUTOLF,A       ;Auto line feed?
0235     20 05                   JR      NZ,UART99      ;No,skip
                         ; Process a line feed
0237     3E 65           LFIN:   LD      A,LF           ;Set LF code
0239     CD 0247         PUT:    CALL    STORE          ;Put it in the buffer
                         ; Exit from the UART poll reset all registers
```

```
 023C     22 2809                    UART99: LD      (INPTR),HL        ;Save pointer
 023F     DD CB 00 D6                        SET     POLL,(IX)         ;Re-enable polling
                                     ;
 0243     E1                         NMIEXT: POP     HL
 0244     F1                                 POP     AF                ;Reset AF
 0245     ED 45                              RETN                      ;Done

                                     ; Put a character into the buffer

 0247     F5                         STORE:  PUSH    AF                ;Save character
 0248     CF                                 RST     INCHL             ;Onto next byte
 0249     7E                                 LD      A,(HL)            ;Check empty
 024A     FE 01                              CP      EMPTY
 024C     28 04                              JR      Z,ROOM            ;Skip if ok
 024E     D7                                 RST     DECHL             ;Else backspace
 024F     DF                                 RST     LEDON             ;Set Error Led
 0250     F1                                 POP     AF                ;Leave
 0251     C9                                 RET
 0252     F1                         ROOM:   POP     AF                ;Set byte
 0253     77                                 LD      (HL),A
 0254     FD 2B                              DEC     IY                ;Update Free count
 0256     FD E5                              PUSH    IY                ;Check it
 0258     E3                                 EX      (SP),HL           ;Get IY - save HL
 0259     7C                                 LD      A,H               ;Check for imminent overflow
 025A     B7                                 OR      A
 025B     20 0B                              JR      NZ,NOTYET
 025D     7D                                 LD      A,L
 025E     FE 05                              CP      5
 0260     30 06                              JR      NC,NOTYET
 0262     DB 30                              IN      A,(SWITCH)        ;Get switch data
 0264     CB B7                              RES     BUSY,A            ;Flag nearly full
 0266     D3 30                              OUT     (SWITCH),A        ;Set it
 0268     E1                         NOTYET: POP     HL                ;Reset HL
 0269     C9                                 RET                       ;Done
                                             SUBTTL  Initialisation
                                             PAGE
 026A

                                 ;   Initialisation routine

 026A    3E C9                   INIT:   LD      A,0C9H          ;Temp ignore NMIs
 026C    32 2800                         LD      (RAM),A
 026F    3E 0F                           LD      A,0FH           ;Ouput mode
 0271    D3 33                           OUT     (PIOBC),A               ;...port B.
 0273    3E CF                           LD      A,0CFH          ;Control mode
 0275    D3 32                           OUT     (PIOAC),A               ;...port A
 0277    3E 1F                           LD      A,MASK          ;I/O mask for..
 0279    D3 32                           OUT     (PIOAC),A               ;...port A
 027B    3E 80                           LD      A,80H           ;Ensure motor off
 027D    D3 31                           OUT     (SOLEN),A
 027F    32 2805                         LD      (HEDDAT),A      ;Clear Head data
 0282    DD 21 2804                      LD      IX,FLAG         ;Initialise IX
 0286    E7                              RST     RESNMI          ;Reset NMI reflection
 0287    DD 36 00 00                     LD      (IX),0          ;Clear flag
 028B    DD 36 FC C3                     LD      (IX+RAM-FLAG),0C3H;Set "JP" instruction
 028F    CD 047E                         CALL    MOVLHS          ;Get head to LHS
 0292    AF                              XOR     A               ;Turn off LED
 0293    CB FF                           SET     RSLED,A
 0295    D3 30                           OUT     (SWITCH),A
                                 ; Now initialise the RAM
 0297    AF                              XOR     A
 0298    32 2833                         LD      (PRTBUF-1),A    ;Set delimiter
 029B    FD 21 037A                      LD      IY,BUFEND-BUFST-1;Set "Free" count in IY
 029F    01 0379                         LD      BC,BUFEND-BUFST-2;Set buffer "empty"
 02A2    11 2886                         LD      DE,BUFST+2
 02A5    21 2884                         LD      HL,BUFST
 02A8    77                              LD      (HL),A          ;Set limit marker
 02A9    23                              INC     HL
 02AA    22 280B                         LD      (PLINE),HL      ;Set O/P pointer
 02AD    36 01                           LD      (HL),EMPTY      ;First marker
 02AF    ED B0                           LDIR                    ;Add the rest
 02B1    12                              LD      (DE),A
                                 ; Set registers for UART
 02B2    21 2BFE                         LD      HL,BUFEND-1     ;Buf start -1
 02B5    22 2809                         LD      (INPTR),HL
                                 ; System now initialised - check for self-test
 02B8    DB 30                           IN      A,(SWITCH)      ;Read switches
 02BA    CB 5F                           BIT     LFSW,A          ;Is LF switch down?
 02BC    20 3A                           JR      NZ,CLRLED       ;No,jump on
                                         SUBTTL  Self test mode
                                         PAGE
```

```
  02BE

                                   ; Self test. Fill buffer with character set

  02BE    11 2885                         LD      DE,BUFST+1       ;Point to start of buffer
  02C1    21 0523                         LD      HL,VERSH         ;Put up version no.
  02C4    01 0018                         LD      BC,VERSHL
  02C7    ED B0                           LDIR                     ;Copy over
  02C9    EB                              EX      DE,HL            ;Now add character set
  02CA    06 04                           LD      B,4              ;Repeat 4 times
  02CC    3E 02                    STEST:  LD      A,2              ;First printing charac.
  02CE    77                       STEST1: LD      (HL),A           ;Store in buffer
  02CF    23                              INC     HL               ;Next byte
  02D0    3C                              INC     A                ;Next character
  02D1    FE 62                           CP      DWON             ;Finished yet?
  02D3    20 F9                           JR      NZ,STEST1        ;No,loop
  02D5    36 65                           LD      (HL),LF          ;Set new line
  02D7    23                              INC     HL
  02D8    3E 02                           LD      A,2              ;Reset A
  02DA    0E 28                    STEST2: LD      C,40             ;Character count
  02DC    36 62                           LD      (HL),DWON        ;Turn on double width
  02DE    23                              INC     HL
  02DF    77                       STEST3: LD      (HL),A           ;Next charac.
  02E0    23                              INC     HL
  02E1    3C                              INC     A
  02E2    0D                              DEC     C                ;More on line?
  02E3    28 F5                           JR      Z,STEST2         ;No,reset Double Width
  02E5    FE 62                           CP      DWON             ;Finished characters?
  02E7    20 F6                           JR      NZ,STEST3        ;No,continue
  02E9    36 65                           LD      (HL),LF          ;Finish this lot
  02EB    23                              INC     HL
  02EC    10 DE                           DJNZ    STEST            ;Repeat if not done
  02EE    2B                              DEC     HL
  02EF    36 67                           LD      (HL),FF          ;Set FF at end
  02F1    22 2809                         LD      (INPTR),HL       ;Update pointer
                                   ; Reset free count in IY
  02F4    FD 21 004E                      LD      IY,BUFEND-BUFST-1-788-VERSHL
                                          SUBTTL  Main routine
                                          PAGE
```

```
 02F8

                                ; Main routine - checks panel switches
                                ;         - Waits until a line is ready and
                                ;         - then formats the line to the print
                                ;         - buffer.
                                ;         - Initiates printing when line ready

 02F8    DB 30                  CLRLED: IN      A,(SWITCH)      ;Turn Off Error LED
 02FA    CB FF                          SET     RSLED,A
 02FC    D3 30                          OUT     (SWITCH),A
 02FE    CD 03F3                MAIN:   CALL    SWSCAN          ;Check the switches
 0301    2A 280B                        LD      HL,(PLINE)      ;Get start of buffer
 0304    CB 7E                          BIT     7,(HL)          ;Is it a graphics line?
 0306    C2 0496                        JP      NZ,GRAFMD       ;Yes,do it
 0309    7E                             LD      A,(HL)          ;Check for empty buffer
 030A    FE 01                          CP      EMPTY           ;Is it?
 030C    28 EA                          JR      Z,CLRLED        ;Yes,turn off LED
 030E    3E                             DEFB    3EH             ;*****Skip the next byte*****
 030F    CF                     MAIN0:  RST     INCHL           ;Onto next byte
 0310    7E                             LD      A,(HL)          ;Get next byte
 0311    FE 01                          CP      EMPTY           ;Empty?
 0313    28 E9                          JR      Z,MAIN          ;Yes,start again
 0315    FE 65                          CP      LF              ;End of line yet?
 0317    38 F6                          JR      C,MAIN0         ;No,move on
                                ; A line is ready - format to buffer
 0319    DD CB 00 FE                    SET     SPACE,(IX)      ;Set "space" flag
 031D    2A 280B                        LD      HL,(PLINE)      ;Start of line
 0320    01 5000                        LD      BC,80*256       ;Set B=80 C=0
 0323    11 2834                        LD      DE,PRTBUF       ;..DE=Print buffer
 0326    7E                     MAIN1:  LD      A,(HL)          ;Get byte
 0327    FE 65                          CP      LF              ;LF or greater?
 0329    30 4F                          JR      NC,MAIN10       ;Yes,off to finish line
 032B    36 01                          LD      (HL),EMPTY      ;Mark empty now
 032D    F5                             PUSH    AF              ;Save A
 032E    CF                             RST     INCHL           ;Advance pointer
 032F    FD 23                          INC     IY              ;Update "Free" count
 0331    F1                             POP     AF              ;Recover A
 0332    FE 62                          CP      DWON            ;Control?
 0334    38 28                          JR      C,PLACE         ;No,put in buffer
                                ; process double width and tab commands
 0336    28 08                          JR      Z,SETDWB        ;Yes,set DW bit
 0338    FE 64                          CP      TAB             ;Tab then?
 033A    28 0C                          JR      Z,DOTAB         ;Yes,do it
 033C    0E 00                          LD      C,0             ;Clear DW bit
 033E    18 E6                          JR      MAIN1
 0340    78                     SETDWB: LD      A,B             ;Check not last charac.
 0341    3D                             DEC     A
 0342    28 E2                          JR      Z,MAIN1         ;Yes,ignore
 0344    0E 80                          LD      C,80H
 0346    18 DE                          JR      MAIN1
                                ; Handle TAB - expand to every 8th column
 0348    3E 50                  DOTAB:  LD      A,80            ;Compute current col.
 034A    90                             SUB     B
 034B    E6 07                          AND     7               ;Mask modulo 8
```

```
  034D     EB                                EX      DE,HL            ;Prt buff to HL
   34E     18 04                             JR      TAB1             ;Skip
  0350     36 02                   TABLP:    LD      (HL),2           ;Set space
  0352     2C                                INC     L                ;Bump address
  0353     05                                DEC     B                ;Update B
  0354     3C                      TAB1:     INC     A                ;More?
  0355     FE 08                             CP      8
  0357     20 F7                             JR      NZ,TABLP         ;Yes,loop
  0359     EB                                EX      DE,HL            ;Reset registers
  035A     3E 02                             LD      A,2              ;Set final space
  035C     18 0E                             JR      PLACSW
                                   ;
  035E     B1                      PLACE:    OR      C                ;Include DW bit if set
  035F     F2 036C                           JP      P,PLACSW         ;Skip if single width
  0362     05                                DEC     B                ;Check OK
  0363     28 04                             JR      Z,NOROOM
  0365     12                                LD      (DE),A           ;Store
  0366     1C                                INC     E
  0367     18 03                             JR      PLACSW
  0369     04                      NOROOM:   INC     B                ;Correct B
   36A     E6 7F                             AND     7FH              ;Clear bit
  036C     12                      PLACSW:   LD      (DE),A           ;Put in Print buffer
  036D     1C                                INC     E
  036E     E6 7F                             AND     7FH              ;Remove bit 7
  0370     FE 02                             CP      2                ;Was it a space?
  0372     28 04                             JR      Z,NOSET          ;Yes,skip
  0374     DD CB 00 BE                       RES     SPACE,(IX)       ;No, flag as printing
  0378     10 AC                   NOSET:    DJNZ    MAIN1            ;Loop if more
                                   ;
                                   ; Line ready so clear remainder of buffer
                                   ;
  037A     22 280B                 MAIN10:   LD      (PLINE),HL       ;Update pointer
  037D     EB                                EX      DE,HL            ;Clear remainder of the buffer
  037E     78                                LD      A,B
  037F     B7                                OR      A                ;See if done
  0380     28 05                             JR      Z,PRINTL
  0382     36 02                   MAIN11:   LD      (HL),2           ;Set to spaces
  0384     2C                                INC     L
   385     10 FB                             DJNZ    MAIN11

                                   ; Print buffer now prepared so start print

  0387     DD CB 00 7E             PRINTL:   BIT     SPACE,(IX)       ;Anything to print?
  038B     20 20                             JR      NZ,LFETC         ;No,skip on
  038D     CD 03E9                           CALL    WBUSY            ;Wait until not busy
  0390     CD 0048                           CALL    MOTON            ;Start motor
  0393     21 008B                           LD      HL,WAIT          ;Set NMI reflection ...
  0396     22 2801                           LD      (NMIREF),HL      ;..for Print wait & go.
  0399     DD CB 00 EE                       SET     PRTING,(IX)      ;Set printing flag

                                   ; Line now being printed, wait for it to stop

  039D     CD 03F3                 PWAIT:    CALL    SWSCAN           ;Scan switches
  03A0     DD CB 00 6E                       BIT     PRTING,(IX)      ;Still printing?
  03A4     20 F7                             JR      NZ,PWAIT         ;Yes,loop
  03A6     DD CB 00 5E                       BIT     BIDIR,(IX)       ;Return head?
```

```
 03AA    CC 0038                   CALL    Z,CHSIDE        ;No,so toggle side

                           ; Line printed - now what about LF etc

 03AD    2A 280B           LFETC:  LD      HL,(PLINE)      ;See where we were
 03B0    7E                        LD      A,(HL)          ;Get the byte
 03B1    FE 65                     CP      LF              ;Not CR/LF/FF?
 03B3    38 1E                     JR      C,ALF           ;Yes,do auto LF
 03B5    36 01             LFETC1: LD      (HL),EMPTY      ;Else flag Empty
 03B7    FD 23                     INC     IY              ;Update "Free" count
 03B9    47                        LD      B,A             ;Temp save in B
 03BA    CF                        RST     INCHL           ;On to next byte
 03BB    78                        LD      A,B             ;Reset
 03BC    22 280B                   LD      (PLINE),HL
 03BF    FE 65                     CP      LF              ;Was it an LF?
 03C1    28 10                     JR      Z,ALF           ;Yes,do it then
 03C3    FE 67                     CP      FF              ;Form Feed then?
 03C5    28 15                     JR      Z,FFF           ;Yes,do it
 03C7    7E                        LD      A,(HL)          ;No,was CR so check next
 03C8    FE 65                     CP      LF              ;This a line feed?
 03CA    28 E9                     JR      Z,LFETC1        ;Yes,tidy buffer and print
 03CC    FE 67                     CP      FF              ;Form feed?
 03CE    28 E5                     JR      Z,LFETC1        ;Yes,tidy etc
 03D0    C3 02FE                   JP      MAIN            ;No,loop
 03D3    CD 03E9           ALF:    CALL    WBUSY           ;Wait until not busy
 03D6    CD 0468                   CALL    SETLF           ;No,start an LF
 03D9    C3 02FE                   JP      MAIN            ;LF initiated,so loop
 03DC    CD 03E9           FFF:    CALL    WBUSY           ;Wait until not busy
 03DF    D9                        EXX                     ;Set for 6 LFs
 03E0    06 90                     LD      B,144
 03E2    D9                        EXX
 03E3    CD 046C                   CALL    GOLF            ;Start it
 03E6    C3 02FE                   JP      MAIN
                                   SUBTTL  Subroutines
                                   PAGE
 03E9

                                 ; Wait while the mechanism is Busy

 03E9    DD CB 00 76             WBUSY:  BIT     BUSY,(IX)         ;Busy?
 03ED    C8                              RET     Z                 ;Return if not
 03EE    CD 03F3                         CALL    SWSCAN            ;Scan the switches
 03F1    18 F6                           JR      WBUSY             ;Loop

                                 ;  Scan Online switch & Line Feed switch

 03F3    DB 30                   SWSCAN: IN      A,(SWITCH)        ;Check switches
 03F5    CB 67                           BIT     ONLINE,A          ;Online?
 03F7    20 21                           JR      NZ,OFFLIN         ;No,say off
 03F9    FD E5                           PUSH    IY                ;Check if busy to clear
 03FB    E3                              EX      (SP),HL           ;Get IY - save HL
 03FC    7C                              LD      A,H
 03FD    B7                              OR      A
 03FE    20 05                           JR      NZ,NOTFUL         ;Ok,skip
 0400    7D                              LD      A,L
 0401    FE 14                           CP      20
 0403    38 06                           JR      C,NEAR            ;Skip if nearly full
 0405    DB 30                   NOTFUL: IN      A,(SWITCH)
 0407    CB F7                           SET     BUSY,A
 0409    D3 30                           OUT     (SWITCH),A
 040B    E1                      NEAR:   POP     HL
 040C    DD CB 00 56                     BIT     POLL,(IX)         ;Were we before?
 0410    20 10                           JR      NZ,CHLFSW         ;Yes,check LF switch
 0412    DB 18                           IN      A,(UARTDT)        ;No,flush Uart
 0414    DD CB 00 D6                     SET     POLL,(IX)         ;Put on line
 0418    18 08                           JR      CHLFSW            ;Check LF switch
 041A    DD CB 00 96             OFFLIN: RES     POLL,(IX)         ;Turn off poll
 041E    CB B7                           RES     BUSY,A
 0420    D3 30                           OUT     (SWITCH),A        ;Set Busy flag
                                 ; Now try the Line Feed switch
 0422    DD CB 00 76             CHLFSW: BIT     BUSY,(IX)         ;Mechanism busy?
 0426    C0                              RET     NZ                ;Yes,don't check
 0427    DB 30                           IN      A,(SWITCH)        ;Get switch
 0429    CB 5F                           BIT     LFSW,A            ;Test switch bit
 042B    C0                              RET     NZ                ;Return if off
 042C    E5                              PUSH    HL                ;Save HL
 042D    C5                              PUSH    BC                ;...and BC
 042E    21 2803                         LD      HL,COUNT          ;Point to count
 0431    36 23                           LD      (HL),35           ;Debounce delay
 0433    7E                      LFD0:   LD      A,(HL)            ;Wait for 15ms
 0434    FE 05                           CP      5
 0436    30 FB                           JR      NC,LFD0
 0438    DB 30                           IN      A,(SWITCH)        ;Check still down
 043A    CB 5F                           BIT     LFSW,A
 043C    20 27                           JR      NZ,LFEXIT         ;Exit if not...
 043E    CD 0468                         CALL    SETLF             ;...else start LF
 0441    DD CB 00 76             LFD1:   BIT     BUSY,(IX)         ;Wait for it to stop
 0445    20 FA                           JR      NZ,LFD1
 0447    06 05                           LD      B,5               ;Initial delay...
 0449    36 FF                   LFD2:   LD      (HL),255          ;...before repeat
```

```
044B    DB 30               LFD3:   IN      A,(SWITCH)
044D    CB 5F                       BIT     LFSW,A          ;Check if still down
044F    20 14                       JR      NZ,LFEXIT       ;Leave if released
0451    7E                          LD      A,(HL)          ;Check delay count
0452    FE 05                       CP      5
0454    30 F5                       JR      NC,LFD3
0456    10 F1                       DJNZ    LFD2
0458    DD CB 00 76         LFD4:   BIT     BUSY,(IX)       ;Doing a line?
045C    CC 0468                     CALL    Z,SETLF         ;Another one if not
045F    DB 30                       IN      A,(SWITCH)
0461    CB 5F                       BIT     LFSW,A
0463    28 F3                       JR      Z,LFD4          ;Loop if still down
                            ; Switch released, reset registers
0465    C1                  LFEXIT: POP     BC              ;Reset BC
0466    E1                          POP     HL              ;and HL
0467    C9                          RET
                            ;
                            ; Start up a line feed
                            ;
0468    D9                  SETLF:  EXX                     ;Get alternate set
0469    06 18                       LD      B,24            ;24 NMIs/line
046B    D9                          EXX
046C    3E 0A               GOLF:   LD      A,10            ;Initial delay
046E    32 2803                     LD      (COUNT),A
0471    DD CB 00 F6                 SET     BUSY,(IX)       ;Now Busy
0475    E5                          PUSH    HL              ;Save HL
0476    21 0074                     LD      HL,LFEED
0479    22 2801                     LD      (NMIREF),HL
047C    E1                          POP     HL              ;Reset HL
047D    C9                          RET

                            ;  Move the head to the LHS if not there

047E    DB 30               MOVLHS: IN      A,(SWITCH)      ;Read switches
0480    CB 57                       BIT     LHLMT,A         ;At lhs?
0482    C8                          RET     Z               ;Yes,return
0483    CD 0048                     CALL    MOTON           ;Start the head moving
0486    DD CB 00 A6                 RES     REVPRT,(IX)     ;Set flag appropriately
048A    DB 30               MOVO:   IN      A,(SWITCH)      ;Wait for it to arrive
048C    CB 57                       BIT     LHLMT,A
048E    20 FA                       JR      NZ,MOVO
0490    DD 36 01 80                 LD      (IX+HEDDAT-FLAG),80H;Stop it
0494    E7                          RST     RESNMI          ;Clear Busy
0495    C9                          RET
                                    SUBTTL   Graphics mode
                                    PAGE
```

```
  0496

                                  ;    Graphics Mode Print

  0496    CD 047E                 GRAFMD: CALL    MOVLHS          ;Get head to LHS
                                  ;
                                  ; Graphics loop - Wait for full graphics line
                                  ;
  0499    CD 03F3                 GLOOP:  CALL    SWSCAN          ;Check the switches
  049C    FD E5                           PUSH    IY              ;Check free count
  049E    E3                              EX      (SP),HL
  049F    11 FF7D                         LD      DE,BUFST-BUFEND+760
  04A2    19                              ADD     HL,DE           ;760 in there yet?
  04A3    E1                              POP     HL              ;(Reset HL)
  04A4    38 F3                           JR      C,GLOOP         ;No,loop
                                  ; A full line of graphics is ready for printing
  04A6    DD CB 00 76                     BIT     BUSY,(IX)       ;Is the mechanism busy?
  04AA    20 ED                           JR      NZ,GLOOP        ;Yes,loop
  04AC    E5                              PUSH    HL              ;Set registers
  04AD    D9                              EXX                     ;..in alternate set
  04AE    E1                              POP     HL
  04AF    01 05F0                         LD      BC,1520         ;NMIs/line
  04B2    D9                              EXX
  04B3    AF                              XOR     A               ;Clear LASTDT
  04B4    32 2806                         LD      (LASTDT),A
  04B7    CD 0048                         CALL    MOTON           ;Start motor
  04BA    21 04D7                         LD      HL,GWAIT        ;Set NMI reflection
  04BD    22 2801                         LD      (NMIREF),HL
  04C0    DD CB 00 EE                     SET     PRTING,(IX)     ;Set printing flag
  04C4    CD 03E9                         CALL    WBUSY           ;Wait until not busy
                                  ; The head is in fact still moving so....
  04C7    21 00AD                         LD      HL,MSTOP
  04CA    22 2801                         LD      (NMIREF),HL
  04CD    DD CB 00 F6                     SET     BUSY,(IX)
  04D1    CD 03E9                         CALL    WBUSY           ;Wait until idle
  04D4    C3 02F8                         JP      CLRLED          ;Do next line,(clear LED)

                                  ; Initial wait for limit switch off

  04D7    F7                      GWAIT:  RST     CHLMSW          ;Check limit switches
  04D8    C2 016F                         JP      NZ,POLLU        ;Still set - so Poll
  04DB    D9                              EXX                     ;Save registers
  04DC    11 04E7                         LD      DE,GPRINT       ;Change reflection
  04DF    ED 53 2801                      LD      (NMIREF),DE
  04E3    D9                              EXX                     ;Restore registers
  04E4    C3 016F                         JP      POLLU

                                  ; Print a graphics line

  04E7    D9                      GPRINT: EXX                     ;Get register set
  04E8    16 00                           LD      D,0             ;Clear D
  04EA    0B                              DEC     BC              ;Update count
  04EB    CB 41                           BIT     0,C             ;Head off?
  04ED    28 15                           JR      Z,GP1           ;Yes,skip
  04EF    7E                              LD      A,(HL)          ;Get next byte
  04F0    E6 7F                       AND     7FH               ;Remove msb
  04F2    36 01                       LD      (HL),EMPTY        ;Flag space empty
  04F4    57                          LD      D,A               ;Temp save in D
  04F5    CF                          RST     INCHL             ;Bump pointer
  04F6    FD 23                       INC     IY                ;...and free space count
  04F8    7A                          LD      A,D               ;Reload
  04F9    DD A6 02                    AND     (IX+LASTDT-FLAG);Check against last
  04FC    28 03                       JR      Z,GP0             ;Skip if no infringement
  04FE    AA                          XOR     D                 ;..Else modify byte to save head
  04FF    57                          LD      D,A
  0500    DF                          RST     LEDON             ;Turn on error LED
  0501    DD 72 02           GP0:     LD      (IX+LASTDT-FLAG),D;Save dot pattern
  0504    DD 72 01           GP1:     LD      (IX+HEDDAT-FLAG),D;Set Head data
  0507    78                          LD      A,B               ;Check for end
  0508    B1                          OR      C
  0509    20 14                       JR      NZ,GP2            ;Ok,carry on
  050B    11 0074                     LD      DE,LFEED          ;Yes,change reflection
  050E    ED 53 2801                  LD      (NMIREF),DE
  0512    06 0E                       LD      B,14              ;Steps/line
  0514    DD 36 FF 01                 LD      (IX+COUNT-FLAG),1;Set delay
  0518    22 280B                     LD      (PLINE),HL        ;Update PLINE
  051B    DD CB 00 AE                 RES     PRTING,(IX)       ;Clear print flag
  051F    D9                 GP2:     EXX                       ;Reset registers
  0520    C3 016F                     JP      POLLU
                                      SUBTTL   Dot patterns
                                      PAGE
  0523

                                   ; First the version number for self-test heading

  0523     64 62 2B 2F             VERSH:  DEFB     TAB,DWON,43,47
  0527     32 54 4B 50                     DEFB     50,84,75,80
  052B     56 02 38 13                     DEFB     86,2,56,19        ;Version 1.0
  052F     10 12 02 13                     DEFB     16,18,2,19        ;Change 6s to 19 & 18
  0533     14 0F 12 15                     DEFB     20,15,18,21
  0537     0F 1A 13 65                     DEFB     15,26,19,LF
  0018                             VERSHL  EQU      $-VERSH
  053B                                     DEFS     560H-$            ;Filler

                                   ; Dot pattern table
  0010                                     .RADIX   16      ;All Hex entries

  0560     00 00 00 00             TABLE:  DEFB     00,00,00,00,00,00,00;
  0564     00 00 00
  0567     00 00 00 7D                     DEFB     00,00,00,7D,00,00,00;!
  056B     00 00 00
  056E     00 70 00 00                     DEFB     00,70,00,00,00,70,00;"
  0572     00 70 00
  0575     09 00 3F 40                     DEFB     09,00,3F,40,09,40,2B;#
  0579     09 40 2B
  057C     10 2A 00 7F                     DEFB     10,2A,00,7F,00,2A,04;$
  0580     00 2A 04
  0583     61 02 64 08                     DEFB     61,02,64,08,13,20,43;%
  0587     13 20 43
  058A     02 25 50 09                     DEFB     02,25,50,09,54,22,01;&
  058E     54 22 01
  0591     00 00 10 20                     DEFB     00,00,10,20,40,00,00;´
  0595     40 00 00
  0598     1C 22 41 00                     DEFB     1C,22,41,00,00,00,00;(
  059C     00 00 00
  059F     00 00 00 00                     DEFB     00,00,00,00,41,22,1C;)
  05A3     41 22 1C
  05A6     22 14 08 77                     DEFB     22,14,08,77,08,14,22;*
  05AA     08 14 22
  05AD     08 00 08 36                     DEFB     08,00,08,36,08,00,08;+
  05B1     08 00 08
  05B4     00 00 0D 02                     DEFB     00,00,0D,02,0C,00,00;,
  05B8     0C 00 00
  05BB     08 00 08 00                     DEFB     08,00,08,00,08,00,08;-
  05BF     08 00 08
  05C2     00 00 03 00                     DEFB     00,00,03,00,03,00,00;.
  05C6     03 00 00
  05C9     01 02 04 08                     DEFB     01,02,04,08,10,20,40;/
  05CD     10 20 40
  05D0     3E 41 04 49                     DEFB     3E,41,04,49,10,41,3E;0
  05D4     10 41 3E
  05D7     00 21 00 7F                     DEFB     00,21,00,7F,00,01,00;1
  05DB     00 01 00
  05DE     21 42 01 44                     DEFB     21,42,01,44,01,48,31;2
  05E2     01 48 31
  05E5     42 01 40 09                     DEFB     42,01,40,09,50,29,46;3
```

```
05E9    50 29 46
05EC    04 08 14 20          DEFB    04,08,14,20,44,1B,04;4
05F0    44 1B 04
05F3    72 01 50 01          DEFB    72,01,50,01,50,01,4E;5
05F7    50 01 4E
05FA    06 09 10 29          DEFB    06,09,10,29,40,09,06;6
05FE    40 09 06
0601    41 02 44 08          DEFB    41,02,44,08,50,20,40;7
0605    50 20 40
0608    36 49 00 49          DEFB    36,49,00,49,00,49,36;8
060C    00 49 36
060F    30 48 01 4A          DEFB    30,48,01,4A,04,48,30;9
0613    04 48 30
0616    00 00 36 00          DEFB    00,00,36,00,36,00,00;:
061A    36 00 00
061D    00 00 6D 02          DEFB    00,00,6D,02,6C,00,00;;
0621    6C 00 C0
0624    00 08 14 22          DEFB    00,08,14,22,41,00,00;<
0628    41 00 00
062B    14 00 14 00          DEFB    14,00,14,00,14,00,14;=
062F    14 00 14
0632    00 00 41 22          DEFB    00,00,41,22,14,08,00;>
0636    14 08 00
0639    20 40 00 45          DEFB    20,40,00,45,08,50,20;?
063D    08 50 20
0640    3E 41 00 59          DEFB    3E,41,00,59,24,41,3C;@
0644    24 41 3C
0647    0F 10 24 40          DEFB    0F,10,24,40,24,10,0F;A
064B    24 10 0F
064E    41 3E 41 08          DEFB    41,3E,41,08,41,08,36;B
0652    41 08 36
0655    3E 41 00 41          DEFB    3E,41,00,41,00,41,22;C
0659    00 41 22
065C    41 3E 41 00          DEFB    41,3E,41,00,41,00,3E;D
0660    41 00 3E
0663    7F 00 49 00          DEFB    7F,00,49,00,49,00,41;E
0667    49 00 41
066A    7F 00 48 00          DEFB    7F,00,48,00,48,00,40;F
066E    48 00 40
0671    3E 41 00 41          DEFB    3E,41,00,41,04,41,26;G
0675    04 41 26
0678    7F 00 08 00          DEFB    7F,00,08,00,08,00,7F;H
067C    08 00 7F
067F    00 41 00 7F          DEFB    00,41,00,7F,00,41,00;I
0683    00 41 00
0686    02 01 00 01          DEFB    02,01,00,01,00,01,7E;J
068A    00 01 7E
068D    7F 00 10 08          DEFB    7F,00,10,08,24,02,41;K
0691    24 02 41
0694    7F 00 01 00          DEFB    7F,00,01,00,01,00,01;L
0698    01 00 01
069B    5F 20 10 08          DEFB    5F,20,10,08,10,20,5F;M
069F    10 20 5F
06A2    5F 20 10 08          DEFB    5F,20,10,08,04,02,7D;N
06A6    04 02 7D
06A9    3E 41 00 41          DEFB    3E,41,00,41,00,41,3E;O
```

```
06AD      00 41 3E
06B0      7F 00 48 00                   DEFB     7F,00,48,00,48,00,30;P
06B4      48 00 30
06B7      3E 41 00 41                   DEFB     3E,41,00,41,04,42,3D;Q
06BB      04 42 3D
06BE      7F 00 48 00                   DEFB     7F,00,48,00,4C,02,31;R
06C2      4C 02 31
06C5      32 49 00 49                   DEFB     32,49,00,49,00,49,26;S
06C9      00 49 26
06CC      40 00 40 3F                   DEFB     40,00,40,3F,40,00,40;T
06D0      40 00 40
06D3      7E 01 00 01                   DEFB     7E,01,00,01,00,01,7E;U
06D7      00 01 7E
06DA      78 04 02 01                   DEFB     78,04,02,01,02,04,78;V
06DE      02 04 78
06E1      7E 01 02 0C                   DEFB     7E,01,02,0C,02,01,7E;W
06E5      02 01 7E
06E8      41 22 14 08                   DEFB     41,22,14,08,14,22,41;X
06EC      14 22 41
06EF      40 20 10 0F                   DEFB     40,20,10,0F,10,20,40;Y
06F3      10 20 40
06F6      41 02 45 08                   DEFB     41,02,45,08,51,20,41;Z
06FA      51 20 41
06FD      00 7F 00 41                   DEFB     00,7F,00,41,00,41,00;[
0701      00 41 00
0704      40 20 10 08                   DEFB     40,20,10,08,04,02,01;\
0708      04 02 01
070B      00 41 00 41                   DEFB     00,41,00,41,00,7F,00;]
070F      00 7F 00
0712      08 10 20 5F                   DEFB     08,10,20,5F,20,10,08;^
0716      20 10 08
0719      01 00 01 00                   DEFB     01,00,01,00,01,00,01;_
071D      01 00 01
0720      00 00 40 20                   DEFB     00,00,40,20,10,00,00;`
0724      10 00 00
0727      02 15 00 15                   DEFB     02,15,00,15,00,14,0B;a
072B      00 14 0B
072E      7F 00 10 01                   DEFB     7F,00,10,01,10,01,0E;b
0732      10 01 0E
0735      0E 00 11 00                   DEFB     0E,00,11,00,11,00,11;c
0739      11 00 11
073C      0E 01 10 01                   DEFB     0E,01,10,01,10,00,7F;d
0740      10 00 7F
0743      0E 01 14 01                   DEFB     0E,01,14,01,14,01,0C;e
0747      14 01 0C
074A      10 00 3F 40                   DEFB     10,00,3F,40,10,40,20;f
074E      10 40 20
0751      09 14 01 14                   DEFB     09,14,01,14,01,14,2B;g
0755      01 14 2B
0758      7F 00 10 00                   DEFB     7F,00,10,00,10,00,0F;h
075C      10 00 0F
075F      00 11 00 5F                   DEFB     00,11,00,5F,00,01,00;i
0763      00 01 00
0766      00 02 01 00                   DEFB     00,02,01,00,01,00,5E;j
076A      01 00 5E
076D      7F 00 04 00                   DEFB     7F,00,04,00,0A,00,11;k
```

Dot patterns

```
 0771       0A 00 11
 0774       00 41 00 7F                    DEFB      00,41,00,7F,00,01,00;l
 0778       00 01 00
 077B       0F 10 08 04                    DEFB      0F,10,08,04,08,10,0F;m
 077F       08 10 0F
 0782       10 0F 00 10                    DEFB      10,0F,00,10,00,10,0F;n
 0786       00 10 0F
 0789       0E 00 11 00                    DEFB      0E,00,11,00,11,00,0E;o
 078D       11 00 0E
 0790       1F 00 04 10                    DEFB      1F,00,04,10,04,10,08;p
 0794       04 10 08
 0797       08 14 00 14                    DEFB      08,14,00,14,00,04,1B;q
 079B       00 04 1B
 079E       10 0F 00 10                    DEFB      10,0F,00,10,00,10,08;r
 07A2       00 10 08
 07A5       08 15 00 15                    DEFB      08,15,00,15,00,15,02;s
 07A9       00 15 02
 07AC       10 00 3E 01                    DEFB      10,00,3E,01,10,01,02;t
 07B0       10 01 02
 07B3       1E 01 00 01                    DEFB      1E,01,00,01,00,1E,01;u
 07B7       00 1E 01
 07BA       18 04 02 01                    DEFB      18,04,02,01,02,04,18;v
 07BE       02 04 18
 07C1       1E 01 02 04                    DEFB      1E,01,02,04,02,01,1E;w
 07C5       02 01 1E
 07C8       11 0A 00 04                    DEFB      11,0A,00,04,00,0A,11;x
 07CC       00 0A 11
 07CF       10 08 05 02                    DEFB      10,08,05,02,00,04,18;y
 07D3       00 04 18
 07D6       11 02 11 04                    DEFB      11,02,11,04,11,08,11;z
 07DA       11 08 11
 07DD       08 00 08 36                    DEFB      08,00,08,36,41,00,41;X
 07E1       41 00 41
 07E4       00 00 00 77                    DEFB      00,00,00,77,00,00,00;X
 07E8       00 00 00
 07EB       41 00 41 36                    DEFB      41,00,41,36,08,00,08;X
 07EF       08 00 08
 07F2       40 00 40 00                    DEFB      40,00,40,00,40,00,40;X
 07F6       40 00 40
 07F9       02 01 00 51                    DEFB      02,01,00,51,08,05,02;X
 07FD       08 05 02
                                           END
```

```
Dot patterns

Macros:

Symbols:
ALF      03D3    AUTOLF   0000    BIDIR    0003    BION     01F8
BSIN     0204    BUFEND   2BFF    BUFST    2884    BUSY     0006
CHLFSW   0422    CHLMSW   0030    CHSIDE   0038    CLRDW    0217
CLRLED   02F8    COUNT    2803    CR       0066    CRIN     0223
DECHL    0010    DOTAB    0348    DOTOFF   0107    DTF      0108
DWIDTH   0001    DWOFF    0063    DWON     0062    ECO      0118
EC1      0122    EMPTY    0001    ENDCHR   010D    ERRMSK   001C
FF       0067    FFF      03DC    FFIN     021F    FLAG     2804
GCOUNT   2807    GLOOP    0499    GOING    01F5    GOLF     046C
GPO      0501    GP1      0504    GP2      051F    GPRINT   04E7
GRAFMD   0496    GRAPH    0000    GWAIT    04D7    HEDDAT   2805
INCHL    0008    INIT     026A    INPTR    2809    LASTDT   2806
LEDON    0018    LF       0065    LFO      0087    LFDO     0433
LFD1     0441    LFD2     0449    LFD3     044B    LFD4     0458
LFEED    0074    LFETC    03AD    LFETC1   03B5    LFEXIT   0465
LFIN     0237    LFSTEP   0005    LFSW     0003    LHLMT    0002
MAIN     02FE    MAINO    030F    MAIN1    0326    MAIN10   037A
MAIN11   0382    MASK     001F    MASKLS   0006    MBS      FFEA
MCR      FFEF    MDWOFF   FFE7    MDWON    FFE6    MFF      FFEE
MLF      FFEC    MOTON    0048    MOTONO   0054    MOTON1   0056
MOVO     048A    MOVLHS   047E    MSTOP    00AD    MSTOPO   00BD
MSTOP1   00CA    MTAB     FFEB    NEAR     040B    NMI      0066
NMIEXT   0243    NMIREF   2801    NOROOM   0369    NOSET    0378
NOTEOL   0131    NOTFUL   0405    NOTYET   0268    NTLO     0145
NTL1     0162    NTL2     016D    OFFLIN   041A    ONGRAF   0001
ONLINE   0004    PIOAC    0032    PIOBC    0033    PLACE    035E
PLACSW   036C    PLINE    280B    POLL     0002    POLLU    016F
PRINT    00D6    PRINTL   0387    PROG     0090    PRT99    016E
PRTBUF   2834    PRTING   0005    PUT      0239    PUTGRF   01E1
PWAIT    039D    RAM      2800    RESET    0000    RESNMI   0020
REVPRT   0004    RHLMT    0001    ROOM     0252    RSLED    0007
SETBI    FFE4    SETDW    0213    SETDWB   0340    SETLF    0468
SETRVP   0043    SETUNI   FFE5    SOLEN    0031    SPACE    0007
STACK    2833    STARTR   00BB    STEST    02CC    STEST1   02CE
STEST2   02DA    STEST3   02DF    STORE    0247    SWIDTH   00FC
SWITCH   0030    SWSCAN   03F3    TAB      0064    TAB1     0354
TABIN    021B    TABLE    0560    TABLP    0350    UARTO    018A
UART99   023C    UARTDT   0018    UARTST   0028    UNION    01FE
VERSH    0523    VERSHL   0018    WAIT     008B    WAITO    00A4
WBUSY    03E9


No  Fatal error(s)
```